

Subgraphs of 4-regular planar graphs

Chris Dowden Louigi Addario-Berry

January 28, 2011

Abstract

We shall present an algorithm for determining whether or not a given planar graph H can ever be a subgraph of a 4-regular planar graph. The algorithm has running time $O(|H|^{2.5})$ and can be used to find an explicit 4-regular planar graph $G \supset H$ if such a graph exists. It shall not matter whether we specify that H and G must be simple graphs or allow them to be multigraphs.

1 Introduction

The last four years have brought large developments in our understanding of random planar graphs. Due to the work of a host of people (see for example [1], [6] and [9]), but in particular the landmark paper by Giménez and Noy [7], the generation of random planar graphs and properties such as the number of edges, average degree, and the number of components are all now well understood. One such result (from [9]) that is related to the subject of this paper is the following: for any fixed planar graph H , the uniform random planar graph P_n on n vertices contains $\Omega(n)$ copies of H except on a set of probability $e^{-\Omega(n)}$.

More recently, Dowden [4] has considered random planar graphs with degree constraints. For example, fix d_1, d_2, D_1 and D_2 and take a graph P_{n,d_1,d_2,D_1,D_2} uniformly at random from the set of all planar graphs on $\{1, 2, \dots, n\}$ with minimum degree between d_1 and d_2 , inclusive, and maximum degree between D_1 and D_2 , inclusive. If $D_2 \geq 3$ (which is necessary for the planarity condition to have any impact), then it is shown that for any fixed connected planar graph H with $\Delta(H) \leq D_2$, P_{n,d_1,d_2,D_1,D_2} also contains $\Omega(n)$ copies of H except on a set of probability $e^{-\Omega(n)}$ (where we ignore odd n if $d_1 = D_2 \in \{3, 5\}$), apart from two special cases. The first is when H is D_2 -regular, in which case the probability is bounded away from both 0 and 1, and the second is when $d_1 = D_2 = 4$ and H happens to be a graph that can *never* be contained within a 4-regular planar graph. This hence raises the question of which graphs can ever be contained in a 4-regular planar graph (we will hereafter refer to such graphs as 4-embeddable), and that is the topic of this paper. (We mention in passing that there is a related body of work on finding minimal regular supergraphs when the planarity restriction is removed; see [2] and the references therein.)

Note that the problem of determining whether or not a given planar graph H can ever be a subgraph of a k -regular planar graph is very straightforward for $k \neq 4$, since the answer is always ‘yes’ if $\Delta(H) \leq k$. This is clear for $k \in \{1, 2\}$, and can easily be proven for $k \in \{3, 5\}$ (note that we must have $k < 6$ for planarity) by showing that in these cases there exist planar graphs that are k -regular except for exactly one vertex with degree $k - 1$, and that we can hence extend H into a k -regular planar graph simply by attaching an appropriate number of these graphs to any vertices of H that have degree less than k . This trick does not work for $k = 4$, however, since clearly a graph that is 4-regular except for exactly one vertex of degree 3 would have to have an odd sum of degrees! In fact, there do actually exist some planar graphs with maximum degree at most 4 that are not 4-embeddable (see Example 1), and so the matter of determining 4-embeddability is non-trivial.

We shall shortly see (in Lemma 2) that a given simple planar graph H is actually 4-embeddable in the world of simple graphs if and only if it is 4-embeddable in the world of multigraphs. Clearly, this second interpretation is just a special case of the more general problem of determining whether or not a given planar *multigraph* H is 4-embeddable. Hence, in this paper we will actually aim to produce an efficient algorithm for the latter problem.

We shall give the details of the algorithm in Section 3. Before this, in Section 2, we will prove three lemmas that shall play important roles. In the first (Lemma 3), we shall observe that if we can extend our multigraph H into a 4-regular planar multigraph, then we can actually do so without introducing any new vertices. In the others (Lemmas 4 and 7), we will show that our problem is straightforward for graphs with a special structure. In Section 3, we shall then give the algorithm itself, which will essentially consist of breaking H up into more and more highly connected pieces until we can apply Lemma 7.

We start with our aforementioned example of a graph with maximum degree 4 that is not 4-embeddable:

Example 1 *No 4-regular planar graph contains a copy of the graph K_5 minus an edge.*

Proof The graph $K_5 - uw$ is drawn with its *unique* planar embedding (see [10]) in Figure 1.

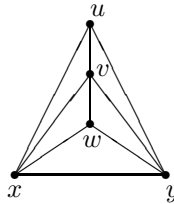


Figure 1: The unique planar embedding of $K_5 - uw$.

Consider any planar graph $G \supset K_5 - uw$ with $\Delta(G) = 4$. Since we already have $\deg_H(v) = \deg_H(x) = \deg_H(y) = 4$, any new edge with at least one end-

point inside the triangle given by vxy must have both endpoints inside. Hence, the sum of degrees inside this triangle must remain odd, and so this region must still contain a vertex of odd degree. Thus, G is not 4-regular. \square

Note that it did not matter whether we took the graph G to be a simple graph or a multigraph. We shall now conclude this introductory section by seeing that this is indeed always the case:

Lemma 2 *Given a simple planar graph H , there exists a 4-regular simple planar graph $G \supset H$ if and only if there exists a 4-regular planar multigraph $G' \supset H$.*

Proof The forward implication is trivial. To see the reverse implication, replace every edge $e = uv$ of $E(G') \setminus E(H)$ by a copy of the graph shown in Figure 2.

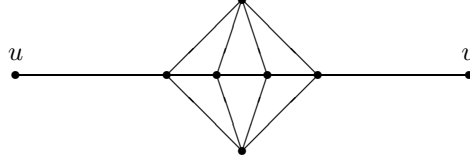


Figure 2: Constructing a 4-regular simple planar graph from a 4-regular planar multigraph.

The resulting graph G will be a 4-regular simple planar graph with $H \subset G$. \square

2 Lemmas and Definitions

In this section, we shall do the groundwork for our algorithm by proving three easy but important lemmas, as well as introducing several helpful definitions. For $v \in V(G)$, $e \in E(G)$, we write $G - v$ for the graph $(V(G) \setminus \{v\}, \{e \in E(G) : v \notin e\})$, and define $G + v$, $G - e$, and $G + e$ similarly.

We start by showing that if there exists a 4-regular planar multigraph $G \supset H$, then we may assume that $V(G) = V(H)$:

Lemma 3 *Given a planar multigraph H , there exists a 4-regular planar multigraph $G \supset H$ if and only if there exists a 4-regular planar multigraph $G' \supset H$ with $V(G') = V(H)$.*

Proof The ‘if’ direction is trivial, so it will suffice to show the ‘only if’ direction.

Suppose there exists a 4-regular planar multigraph $G \supset H$, and let G' be a *minimal* such graph, in the sense that $|V(G') \setminus V(H)|$ is as small as possible.

Suppose $|V(G') \setminus V(H)| \neq 0$ (hoping to obtain a contradiction) and let $v \in V(G') \setminus V(H)$. We shall show that we can obtain a 4-regular planar multigraph G^* such that $H \subset G^*$ and $V(G^*) = V(G') \setminus v$, thus obtaining our desired contradiction:

Case (a): If v has two loops to itself, then we may simply take G^* to be $G' - v$.
Case (b): If v has exactly one loop to itself and its other two neighbours are v_1 and v_2 (where we allow the possibility that $v_1 = v_2$), then we may take G^* to be $(G' - v) + v_1v_2$.

Case (c): If v has no loops to itself, then fix a plane drawing of G' and let e_1, e_2, e_3 and e_4 be the edges incident to v in clockwise order in this plane drawing. Let v_1, v_2, v_3 and v_4 , respectively, denote the other endpoints of e_1, e_2, e_3

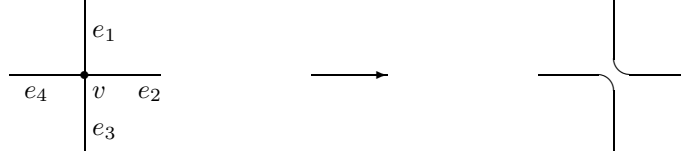


Figure 3: Constructing a smaller 4-regular planar multigraph in case (c).

and e_4 (allowing the possibility that $v_i = v_j$ for some i and j). Then we may take G^* to be $(G' - v) + v_1v_2 + v_3v_4$, since this can also be drawn in the plane (see Figure 3). \square

Note that Lemma 3 itself provides a way to determine algorithmically whether or not a given planar multigraph H can ever be a subgraph of a 4-regular planar multigraph, since it will suffice just to check all 4-regular planar multigraphs with the same vertex set as H . However, we shall produce a much faster algorithm in Section 3. The following lemma is a step in that direction, establishing a polynomial time algorithm for the case when H is 3-vertex-connected:

Lemma 4 *Given any 3-vertex-connected planar multigraph H , we can determine in $O(|H|^{2.5})$ operations whether or not H is 4-embeddable.*

Proof Without loss of generality, $\Delta(H) \leq 4$. Thus, 3-vertex-connectivity implies that H has no loops. By a result of Whitney [10] on 3-vertex-connected simple graphs, it then follows that H has a unique planar embedding¹, and this can be obtained in $O(|H|)$ operations (see [3]). We shall use this embedding to reduce the problem of 4-embeddability to finding a perfect matching in a suitably defined ‘auxiliary’ graph.

Let $f(v) = 4 - \deg_H(v) \forall v \in V(H)$. Then we define our auxiliary graph A (which will not necessarily be planar) to consist of $f(v)$ copies, $v_1, v_2, \dots, v_{f(v)}$, of each vertex v , with an edge between v_i and w_j precisely if v and w are on a common face in H (here we allow $v = w$, but not $v_i = v_j$). We claim that H is 4-embeddable if and only if A has a perfect matching.

¹By “unique” we mean “unique up to automorphisms of plane graphs”. In particular, the presence of parallel edges cannot affect the uniqueness of the embedding.

First, suppose H is 4-embeddable, i.e. $H \subset G$ for some 4-regular planar multigraph G . By Lemma 3, we may assume that $V(G) = V(H)$, in which case the edges $E(G) \setminus E(H)$ form a perfect matching in A (where we choose which copy of v to use for a given edge in some arbitrary consistent manner). Conversely, if we can find a perfect matching in A , then inserting the edges of this matching into our embedding of H will give us a (not necessarily plane) 4-regular multigraph, which can then be made into a plane 4-regular multigraph simply by separating any new edges that cross, as in Figure 4.

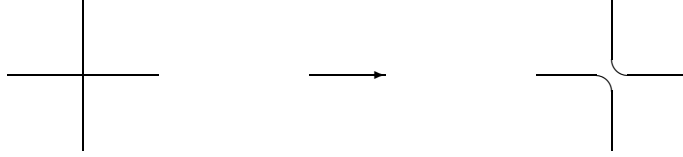


Figure 4: Separating crossing edges of our matching.

Note that we can obtain the auxiliary graph in $O(|H|^2)$ time, since there are $O(|H|^2)$ possible edges. This graph will have at most $4|H|$ vertices, and so we can then determine whether or not it has a perfect matching in $O(|H|^{2.5})$ time (see [5]). \square

We remark that the only place in the preceding proof where we used 3-vertex-connectivity was to ensure that H had a unique planar embedding. Thus, an identical proof gives a polynomially testable necessary and sufficient condition for any given *plane* graph to be a subgraph of a 4-regular planar graph. However, note that this condition does not yield an efficient algorithm for an arbitrary *planar* graph H , as in general H may have exponentially many planar embeddings.

Returning to our main thrust, recall that in the proof of Lemma 4 we introduced the function f , which encoded the discrepancy between a vertex's degree in H and its target degree (in this case, there was a target degree of 4 for all vertices). Such functions will play an important role in our algorithm, and so we will now take the time to set up a more general framework for them:

Definition 5 *Given a planar multigraph H with maximum degree at most 4, $f_H : V(H) \rightarrow \mathbf{N}$ is a discrepancy function on H if (a) $f_H(v) \leq 4 - \deg_H(v) \forall v \in V(H)$ (we call this the discrepancy inequality) and (b) $\sum_{v \in V(H)} f_H(v)$ is even (we call this discrepancy parity). If it is also the case that $f_H(v) + \deg_H(v)$ is even for all $v \in V(H)$, we call f_H an even discrepancy function on H .*

We say that a plane multigraph G satisfies (H, f_H) if $V(G) = V(H)$, $E(G) \supset E(H)$ and $\deg_G(v) = \deg_H(v) + f_H(v) \forall v$. If such a plane multigraph G exists, we say that f_H can be satisfied on H , or that (H, f_H) can be satisfied.

We next also introduce ‘augmentations’, a graph operation we will use repeatedly during the algorithm:

Definition 6 Given a multigraph B , we define the operation of placing a diamond on an edge $uv \in E(B)$ to mean that we subdivide the edge with three vertices and then also add two other new vertices so that they are both adjacent to precisely these three vertices. We define the operation of placing a vertex on an edge $xy \in E(B)$ to mean that we subdivide the edge with a single vertex.

Given multigraphs B and R and a discrepancy function f_R , we say that (R, f_R) is an augmentation of B if R can be formed from B by placing vertices and diamonds on some of the edges of B (in such a way that there is at most one vertex or diamond on each original edge) and if $f_R = 4 - \deg_R$ for all vertices in the new diamonds and $f_R \in \{1, 2\}$ for the other new vertices.

An example of an augmentation is given in Figure 5. When we break H up into pieces in our algorithm, the augmentation of a piece will capture the key information about how it interacted with the rest of H .

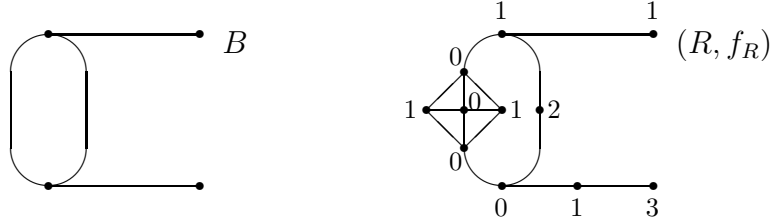


Figure 5: A planar multigraph and an augmentation of it.

Note that in performing such an augmentation, we may differently augment two parallel edges (as in Figure 5). This causes an asymmetry in the graph, and so R may not have a unique planar embedding even if B does. However, augmenting two parallel edges in the *same* way will not affect the uniqueness of the embedding. We will use this fact in Lemma 7, below, which extends Lemma 4 to cover augmentations of 3-vertex-connected multigraphs:

Lemma 7 Let B be a planar multigraph of maximum degree at most 4 that contains no 2-vertex-cuts (so $|B| \leq 3$ or B is 3-vertex-connected), and let (R, f_R) be an augmentation of B . Suppose we know which parts of R correspond to which edges of B . Then $\exists \lambda$, independent of B and R , such that we can determine in at most $\lambda|B|^{2.5}$ operations whether or not (R, f_R) can be satisfied.

Proof Without loss of generality, we may assume that $|B| > 3$ (since if B is bounded then there are only a finite number of possibilities for (R, f_R) , and the satisfiability of these can be determined in finite time). Thus, as in Lemma 4, B has no loops and so has a unique planar embedding. Hence, R will also have a unique planar embedding, apart from possibly at places where B has multi-edges.

Note that all vertices in B must have at least 3 distinct neighbours, since B does not contain any 2-vertex-cuts. Hence (since $\deg_B(x) = \deg_R(x) \leq 4 - f_R(x) \forall x \in V(B)$), if vertices u and v have a multi-edge between them in B , then it must be only a double-edge and it must be that $f_R(u) = f_R(v) = 0$.

We shall now use this information to find a pair $(R', f_{R'})$ such that R' has a unique planar embedding and (R, f_R) can be satisfied if and only if we can satisfy $(R', f_{R'})$.

Let Type A, Type B, Type C and Type D denote the four possible ‘augmented versions’ of an edge, as shown in Figure 6, and recall that R will have a

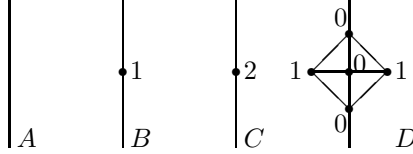


Figure 6: Augmented versions of an edge.

unique embedding apart from at any places where B has a double-edge. More strongly, the only possible non-uniqueness arises when B has a double-edge and these edges are of distinct types in the augmentation. (This is explained in the paragraph just before the statement of the lemma.) We now explain how to deal with such situations. If there exist vertices u and v with a Type A-Type D double-edge between them, then it can be seen that it is impossible to satisfy (R, f_R) , since $f(u) = f(v) = 0$ (see Figure 7). If we have no Type A-Type D

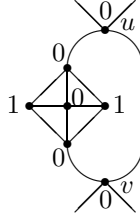


Figure 7: A Type A-Type D double edge.

double-edges, then let R' be formed from R as follows:

- (i) If the augmented versions of a double-edge are Type A and Type B, then delete the Type A part;
- (ii) If A and C, delete C;
- (iii) If B and C, delete C;
- (iv) If B and D, delete D.
- (v) If C and D, delete C.

Let $f_{R'}(v) = f_R(v) \forall v \in V(R')$.

Using the fact that the two ends of any double-edge must have $f_R = 0$, it is easy to see that (R, f_R) can be satisfied if and only if $(R', f_{R'})$ can be satisfied. It is also clear that R' will have a unique embedding (if we consider the vertices as unlabelled apart from the discrepancy function). Thus, to determine whether or not $(R', f_{R'})$ can be satisfied, it suffices to see if we can satisfy $(R', f_{R'})$ in this embedding.

Note that R' can be generated from R in $O(|B|)$ time, since there are $O(|B|)$ edges in B and we know which parts of R correspond to which edges of B . Since $|R'| = O(|B|)$, the planar embedding of R' can then be found in $O(|B|)$ time (see [3]), and the satisfiability of $(R', f_{R'})$ can then be determined in $O(|B|^{2.5})$ operations by looking for a perfect matching in the corresponding auxiliary graph, as in the proof of Lemma 4. \square

It is worth remarking that, apart from obtaining the auxiliary graph and looking for a perfect matching, all the procedures involved in the previous proof actually only take $O(|B|)$ time.

3 The Algorithm

We will now present our algorithm. We shall first provide a short sketch, before then giving the details in full. Afterwards, we will investigate the running time.

Sketch of Algorithm

The algorithm shall consist of four stages, each of which will involve breaking H up into more highly connected pieces, until we can eventually apply Lemma 7 to all of these.

We will start, in Stages 1 and 2, by straightforwardly showing that H is 4-embeddable if and only if all its 2-edge-connected components are. This part of our argument will not require the use of either Lemma 3 or Lemma 7.

In Stage 3, we will break our 2-edge-connected components into 2-vertex-connected blocks, and show that the discrepancy function $f = 4 - \deg$ can be satisfied on our 2-edge-connected components if and only if certain specified discrepancy functions can be satisfied on all the 2-vertex-connected blocks. We shall use Lemma 3 to simplify our arguments here.

Stage 4 is where we will use the notion of augmentations. We shall split our 2-vertex-connected blocks into 3-vertex-connected multigraphs and define augmented versions of each of these. There will be different cases depending on exactly how the 2-vertex-cuts break up the graph, and we will show that the discrepancy functions defined on our 2-vertex-connected blocks can be satisfied if and only if all these augmentations can be satisfied. This can then be determined using Lemma 7.

FULL ALGORITHM

STAGE 1

Clearly, there exists a 4-regular planar multigraph $G \supset H$ if and only if there exist 4-regular planar multigraphs $G_i \supset H_i$ for all components H_i of H (the ‘if’

direction follows by taking G to be the graph whose components are the G_i 's and the 'only if' direction follows by taking $G_i = G \forall i$.

Thus, the first stage of our algorithm will be to split H into its components.

STAGE 2

Let H_1 be a component of H and suppose that H_1 has a cut-edge $e = uv$. Let H_u and H_v denote the components of $H_1 - e$ containing u and v , respectively. Clearly, there exists a 4-regular planar multigraph $G_1 \supset H_1$ only if there exist 4-regular planar multigraphs $G_u \supset H_u$ and $G_v \supset H_v$ (this follows by taking $G_u = G_v = G_1$). We shall now see that the converse is also true:

Suppose there exist 4-regular planar multigraphs $G_u \supset H_u$ and $G_v \supset H_v$. Note that $\deg_{H_u}(u) = \deg_{H_1}(u) - 1 \leq 3$, since $v \notin V(H_u)$, so $\exists w \in V(G_u)$ such that $uw \in E(G_u) \setminus E(H_u)$. (It is possible that $w = u$.) Similarly, $\exists x \in V(G_v)$ such that $vx \in E(G_v) \setminus E(H_v)$. Since G_u and G_v are both planar, they can be drawn with the edges uw and vx , respectively, in the outside face. Thus, the graph G_1 formed by deleting these two edges and inserting edges uv and wx will also be planar, as well as being a 4-regular multigraph containing H_1 (see Figure 8).

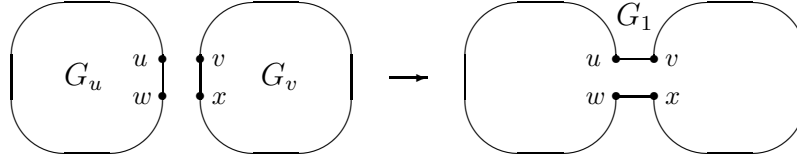


Figure 8: Constructing a 4-regular planar multigraph G_1 from 4-regular planar multigraphs G_u and G_v .

We have shown that H_1 is 4-embeddable if and only if H_u and H_v both are. Thus, by repeated use of this result, we find that H_1 is 4-embeddable if and only if all its 2-edge-connected components are (counting an isolated vertex as 2-edge-connected).

Therefore, the second stage of our algorithm will be to split the components of H into their 2-edge-connected components.

STAGE 3

Let A be one of our 2-edge-connected components. We wish to determine whether or not there exists a 4-regular planar multigraph $G_A \supset A$. By Lemma 3, it suffices to discover whether or not there exists a 4-regular planar multigraph $G' \supset A$ with $V(G') = V(A)$, i.e. to determine whether or not we can satisfy the even discrepancy function on A defined by setting $f_A(v) = 4 - \deg_A(v) \forall v \in V(A)$.

Suppose that A has a cut-vertex v . Since A contains no cut-edges, it must be that $A - v$ consists of exactly two components, A_1 and A_2 , with exactly two edges from v to each of these components. Thus, $\deg_A(v) = 4$ and $f_A(v) = 0$.

Let A_1^* denote the planar multigraph induced by $V(A_1) \cup v$ and let $f_{A_1^*}$ denote the even discrepancy function on A_1^* defined by setting $f_{A_1^*}(x) = f_A(x) \ \forall x \in V(A_1^*)$ (we have $f_{A_1^*}(x) + \deg_{A_1^*}(x) = 4 \ \forall x \neq v$ and $f_{A_1^*}(v) + \deg_{A_1^*}(v) = 2$, so $f_{A_1^*}$ is indeed an even discrepancy function). Let A_2^* and $f_{A_2^*}$ be defined similarly (see Figure 9).

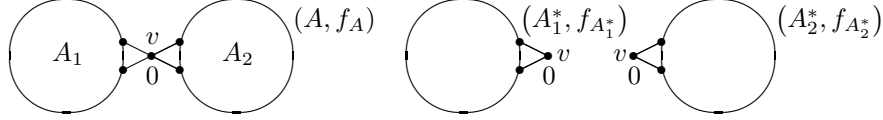


Figure 9: The planar multigraphs A, A_1^* and A_2^* .

Clearly, we can satisfy (A, f_A) if we can satisfy both $(A_1^*, f_{A_1^*})$ and $(A_2^*, f_{A_2^*})$ (since if there exist plane multigraphs G_1^* and G_2^* satisfying $(A_1^*, f_{A_1^*})$ and $(A_2^*, f_{A_2^*})$, respectively, then we may assume that v is in the outside face of both of these, and so we can then ‘glue’ these two drawings together at v to obtain a plane multigraph that satisfies (A, f_A)). We shall now see that the converse is also true:

Suppose (A, f_A) can be satisfied, i.e. there exists a plane multigraph $G' \supset A$ with $V(G') = V(A)$ and $\deg_{G'}(x) = 4 \ \forall x$. Let us consider the induced plane drawing of A . Since A_2 is connected, it must lie in a single face of A_1^* . Thus, we may assume that our plane drawing of A is as shown in Figure 10, where without loss of generality we have drawn A_2 in the outside face of A_1^* . Note that the set of edges in $E(G') \setminus E(A)$ between A_1 and A_2 must all lie in a single face of our plane drawing and that there must be an even number of such edges, since f_A is an even discrepancy function and $f_A(v) = 0$. Thus, we may ‘pair up’ these edges, as in Figure 10, to obtain a plane multigraph G^* satisfying (f_A, A) that has *no* edges from A_1 to A_2 . It is then clear that $G_1^* = G^* \setminus A_2$ and $G_2^* = G^* \setminus A_1$ will satisfy $(A_1^*, f_{A_1^*})$ and $(A_2^*, f_{A_2^*})$, respectively.

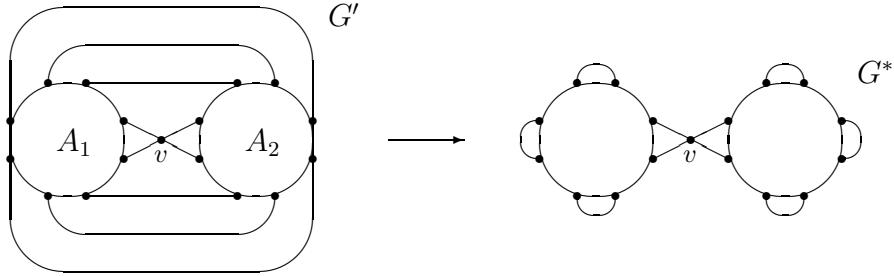


Figure 10: Constructing the graph G^* from G' .

Thus, we have shown that the even discrepancy function f_A can be satisfied on A if and only if the even discrepancy functions $f_{A_1^*}$ and $f_{A_2^*}$ can be satisfied on A_1^* and A_2^* , respectively. By repeatedly using this result, we may obtain a

set of discrepancy functions defined on 2-vertex-connected planar multigraphs such that (A, f_A) can be satisfied if and only if all these can be satisfied.

Therefore, the third stage of our algorithm will be to split our 2-edge-connected components into 2-vertex-connected blocks (the decomposition is, in fact, unique), and give each the appropriate discrepancy function.

STAGE 4

Let C be one of our 2-vertex-connected blocks. We wish to determine whether or not (C, f_C) can be satisfied. Analogously to Stages 1-3, we shall split C up into pieces at 2-vertex-cuts. However, unlike with these earlier stages, this time if there exists a graph M satisfying (C, f_C) there may be several different possibilities for how the edges of M could interact with these pieces. To keep track of this, we shall define augmentations of the pieces in such a way that (C, f_C) can be satisfied if and only if these augmentations can all be satisfied.

We will proceed iteratively. At the start of each iteration, we shall have a ‘blue’ graph (which will initially be C) and an augmentation of it (initially (C, f_C)) for which we want to determine satisfiability (we shall refer to this augmentation as a ‘red’ graph with a discrepancy function). We will split our blue graph in two at a 2-vertex-cut by breaking off a 3-vertex-connected piece, and we shall define augmentations of these two pieces in such a way that the augmentation of the blue graph can be satisfied if and only if the augmentations of the pieces can. Lemma 7 can then be used to determine satisfiability of the augmentation of the 3-vertex-connected piece, while the other piece and its augmentation can be used as the inputs for the next iteration. The iterative loop terminates when the blue graph is itself 3-vertex-connected.

We shall now give the full details:

Initialising

Let us define our initial ‘blue graph’, B , to be C , let us also define our initial ‘red graph’, R , to be C , and let R have discrepancy function $f_R = f_C$. Note that (R, f_R) is an augmentation of B . At the start of each iteration, we will always have a blue planar multigraph with no cut-vertex, and an augmentation of this consisting of a red graph and a discrepancy function.

The Iterative Loop

Check if B has any 2-vertex-cuts. If not, then we are done, since we can simply use Lemma 7. Otherwise, let us find a minimal 2-vertex-cut $\{u, v\}$, where we use ‘minimal’ to mean that the component of smallest order in $B - u - v$ is minimal over all possible 2-vertex-cuts.

We shall now proceed to define several graphs based on the pieces of $B - u - v$ (these definitions are illustrated in Figure 11). Let B_1 denote a component of smallest order in $B - u - v$, let B_1^* denote the graph induced by $V(B_1) \cup \{u, v\}$ and let B_1^\dagger denote the graph obtained from B_1^* by deleting any edges from u to

v . Let $B_2 = B \setminus B_1^*$, let $B_2^* = B \setminus B_1$ and let B_2^\dagger denote the graph obtained from B_2^* by deleting any edges from u to v . Let $R_1^*, R_2^*, R_1^\dagger$ and R_2^\dagger , respectively, denote the red versions of $B_1^*, B_2^*, B_1^\dagger$ and B_2^\dagger that follow ‘naturally’ from R , and let $R_1 = R \setminus R_2^*$ and $R_2 = R \setminus R_1^*$.

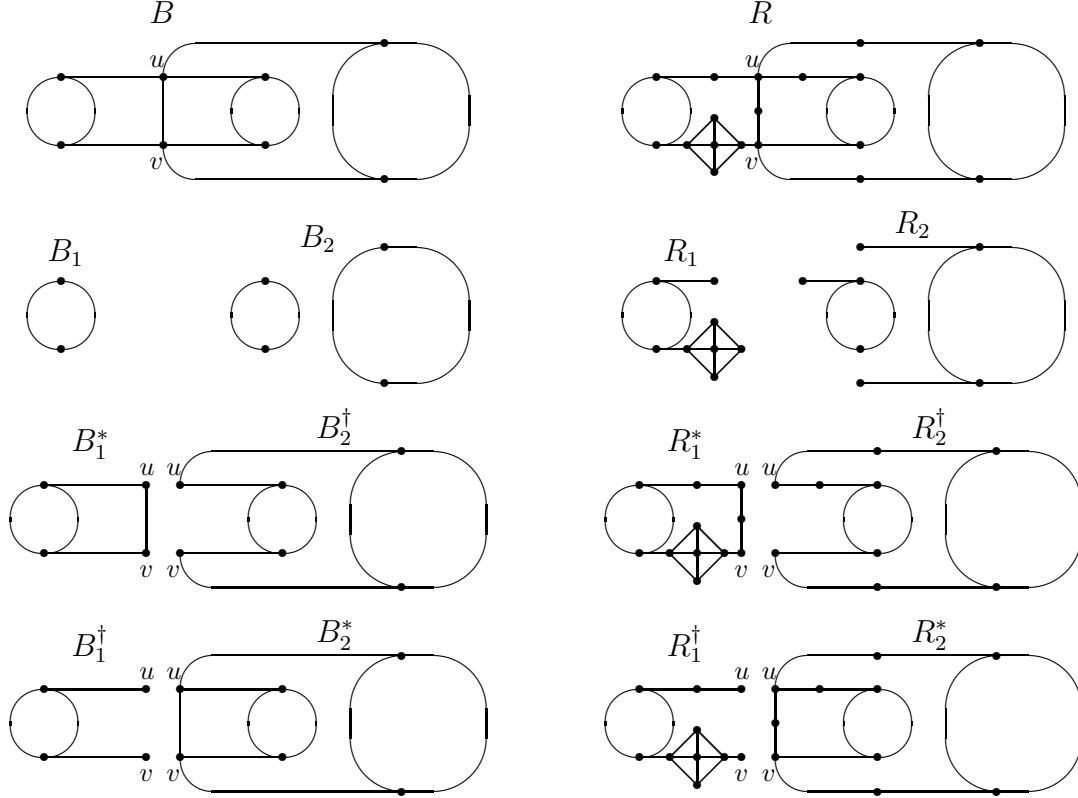


Figure 11: The planar multigraphs defined in the iterative loop of Stage 4.

Let $u1$ denote the statement

$$'f_R(u) = 0 \text{ or there is only one edge in } B \text{ from } u \text{ to } B_1'$$

(note that the latter clause of $u1$ implies $|B_1| = 1$, by the minimality of $\{u, v\}$, but that it is not equivalent to this, as we may have multi-edges). It is important to note that the number of edges in B from u or v to B_1 is exactly the same as the number of edges in R from u or v , respectively, to R_1 (and similarly for B_2 and R_2). Thus, $u1$ is equivalent to the statement ' $f_R(u) = 0$ or there is only one edge in R from u to R_1 '. Let $v1$ denote the analogous statement to $u1$ for v , and let $u2$ and $v2$ denote the analogous statements for B_2 . Let $\overline{u1}, \overline{v1}, \overline{u2}$ and $\overline{v2}$ denote the complements of $u1, v1, u2$ and $v2$.

Recall that we wish to split our graph in two at each iteration. Note that if we have $\overline{u2}$, for example, then $f_R(u) \geq 1$ and there are at least two edges in R

from u to R_2 , so there may be several possibilities for where a graph satisfying (R, f_R) could have a new $u-R_2$ edge. This could complicate matters, causing an exponential blow-up in the running time, unless we choose to split the graph in such a way that only the edges from u to R_1 are important to the analysis. Thus, our choice of how best to split the graph depends on which of the statements $u1, v1, u2$ and $v2$ are true, and hence our next step is to divide our iterative loop into different cases based on this information.

Case (a): $u1 \wedge v1$

We shall first establish a couple of important facts, before then splitting into two further subcases arising from parity issues.

Note that, by definition, B_1 is connected. Thus, R_1 must also be connected, and so has to lie in a single face of R_2^* . Hence, in any planar embedding R must look as in Figure 12, where broken lines represent edges that may or may not exist and where, without loss of generality, we have drawn R_1 in the outside face of R_2^* . Therefore, if a plane multigraph M satisfies (R, f_R) then all edges

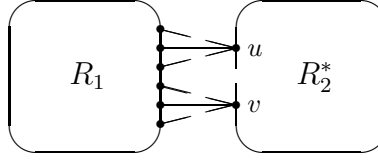


Figure 12: The planar multigraph R .

in $E(M) \setminus E(R)$ between $V(R_1)$ and $V(R_2^*)$ must lie within only two faces of the induced embedding of R (since, by the definition of $u1$, u can have more than one edge to R_1 only if $f(u) = 0$, and similarly for v).

Secondly, since f_R satisfies discrepancy parity, note that $\sum_{x \in V(R_1)} f_R(x)$ and $\sum_{x \in V(R_2^*)} f_R(x)$ must either both be odd or both be even.

Case (a)(i): $\sum_{x \in V(R_1)} f_R(x)$ and $\sum_{x \in V(R_2^*)} f_R(x)$ both odd

Let $B'_1 = B_1^\dagger + uv$ and let $B'_2 = B_2^* + uv$ (so uv will now be a multi-edge in B'_2 if $uv \in E(B)$). We shall now define an augmentation $(R'_1, f_{R'_1})$ of B'_1 and an augmentation $(R'_2, f_{R'_2})$ of B'_2 such that (R, f_R) can be satisfied if and only if $(R'_1, f_{R'_1})$ and $(R'_2, f_{R'_2})$ can both be satisfied (these new augmentations are illustrated in Figure 13).

Let R'_1 be the graph formed from R_1^\dagger by relabelling u and v as u_1 and v_1 , respectively, and introducing a new vertex w_1 with edges to both u_1 and v_1 . Similarly, let R'_2 be the graph formed from R_2^* by relabelling u and v as u_2 and v_2 , respectively, and introducing a new vertex w_2 with edges to both u_2 and v_2 . Let $f_{R'_1}$ be the discrepancy function on R'_1 defined by setting $f_{R'_1}(u_1) = f_{R'_1}(v_1) = 0$, $f_{R'_1}(w_1) = 1$, and $f_{R'_1}(x) = f_R(x) \forall x \in V(R_1)$. Let $f_{R'_2}$ be the discrepancy function on R'_2 defined by setting $f_{R'_2}(u_2) = f_R(u)$, $f_{R'_2}(v_2) = f_R(v)$, $f_{R'_2}(w_2) =$

1, and $f_{R'_2}(x) = f_R(x) \forall x \in V(R_2)$. (Note that $f_{R'_1}$ and $f_{R'_2}$ are both valid discrepancy functions, since the discrepancy inequality is clearly satisfied by both and discrepancy parity follows from the facts that $\sum_{x \in V(R'_1)} f_{R'_1}(x) = \sum_{x \in V(R_1)} f_R(x) + 1$, that $\sum_{x \in V(R'_2)} f_{R'_2}(x) = \sum_{x \in V(R_2^*)} f_R(x) + 1$ and that $\sum_{x \in V(R_1)} f_R(x)$ and $\sum_{x \in V(R_2^*)} f_R(x)$ are both odd).

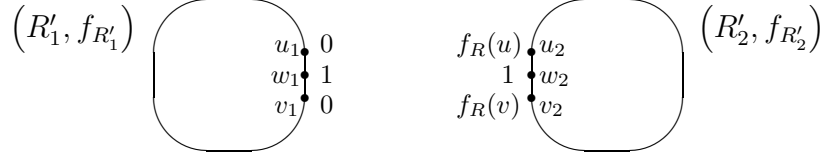


Figure 13: The planar multigraphs R'_1 and R'_2 , with their discrepancy functions.

Claim 8 (R, f_R) can be satisfied if and only if $(R'_1, f_{R'_1})$ and $(R'_2, f_{R'_2})$ can both be satisfied.

Proof Suppose first that a plane multigraph M satisfies (R, f_R) . Since $\sum_{x \in V(R_1)} f_R(x)$ and $\sum_{x \in V(R_2^*)} f_R(x)$ are both odd, there must be an odd number of edges in $E(M) \setminus E(R)$ between $V(R_1)$ and $V(R_2^*)$. As already noted, these edges must all lie within two faces of the embedding of R induced from M . Thus, one of these faces must have an odd number of new edges and the other must have an even number. By pairing edges up, as in the second half of Stage 3, we can hence obtain a planar multigraph satisfying (R, f_R) that has exactly one new edge between $V(R_1)$ and $V(R_2^*)$. It is then easy to see that we can satisfy both $(R'_1, f_{R'_1})$ and $(R'_2, f_{R'_2})$.

Suppose next that $(R'_1, f_{R'_1})$ and $(R'_2, f_{R'_2})$ can both be satisfied, by plane multigraphs $M_{R'_1}$ and $M_{R'_2}$ respectively, and let the edges adjacent to w_1 in $E(M_{R'_1}) \setminus E(R'_1)$ and w_2 in $E(M_{R'_2}) \setminus E(R)$ be denoted by $e_1 = z_1 w_1$ and $e_2 = z_2 w_2$ respectively. We may assume that e_1 is in the outside face of $M_{R'_1}$. Note that the edges $u_1 w_1$ and $v_1 w_1$ must then be in the outside face of $M_{R'_1} - e_1$, since these are the only edges incident to w_1 in $M_{R'_1} - e_1$. Hence, by turning our drawing upside-down if necessary, we may assume that u_1, w_1 and v_1 are in clockwise order around this outer face of $M_{R'_1} - e_1$, and so $M_{R'_1}$ is as shown in Figure 14 (where, without loss of generality, we have drawn e_1 so that v_1 is also in the outside face of $M_{R'_1}$). Similarly, we may assume that $M_{R'_2}$ is also as shown in Figure 14. It is then clear that we can delete w_1 and w_2 , ‘glue’ u_1 to u_2 and v_1 to v_2 (i.e. identify u_1 and u_2 and, separately, v_1 and v_2), and insert the edge $z_1 z_2$ to obtain a plane multigraph M_R that will satisfy (R, f_R) (note that it doesn’t matter whether or not $z_2 \in \{u_2, v_2\}$). \square

Recall that $B'_1 = B_1^\dagger + uv$ and note that B'_1 must not contain any 2-vertex-cuts, by the minimality of B_1 . Thus, by Lemma 7, in $O(|B'_1|^{2.5})$ time we can determine whether or not $(R'_1, f_{R'_1})$ can be satisfied. If it cannot, we terminate

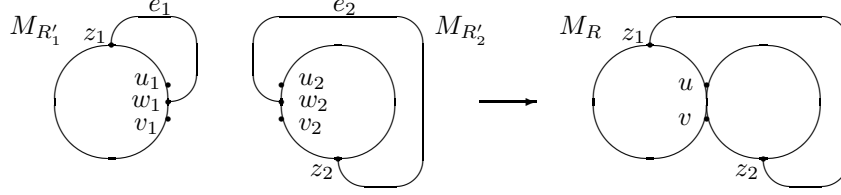


Figure 14: Constructing a planar multigraph M_R satisfying (R, f_R) .

the algorithm. If it can, we return to the start of the iterative loop with B'_2 as our new blue graph, R'_2 as our new red graph and $f_{R'_2}$ as our new discrepancy function (note that, as required, B'_2 does not contain a cut-vertex since otherwise this would also be a cut-vertex in B — this property will be required for case (b)).

Case (a)(ii): $\sum_{x \in V(R_1)} f_R(x)$ and $\sum_{x \in V(R_2^*)} f_R(x)$ both even

Again, we let $B'_1 = B_1^\dagger + uv$ and $B'_2 = B_2^* + uv$. This time, we shall define augmentations $(R'_1, f_{R'_1})$ and $(R''_1, f_{R''_1})$ of B'_1 and augmentations $(R'_2, f_{R'_2})$, $(R''_2, f_{R''_2})$ and $(R'''_2, f_{R'''_2})$ of B'_2 (see Figure 15) such that (R, f_R) can be satisfied if and only if:

- (1) $(R'_1, f_{R'_1})$ and $(R'_2, f_{R'_2})$ can both be satisfied, but $(R''_1, f_{R''_1})$ can't;
- (2) $(R''_1, f_{R''_1})$ and $(R''_2, f_{R''_2})$ can both be satisfied, but $(R'_1, f_{R'_1})$ can't; or
- (3) $(R'_1, f_{R'_1})$, $(R''_1, f_{R''_1})$ and $(R'''_2, f_{R'''_2})$ can all be satisfied.

Let R'_1 be the graph formed from R_1^\dagger by relabelling u and v as u_1 and v_1 , respectively, and inserting an edge between u_1 and v_1 . Let $f_{R'_1}$ be the discrepancy function on R'_1 defined by setting $f_{R'_1}(u_1) = f_{R'_1}(v_1) = 0$ and $f_{R'_1}(x) = f_R(x) \forall x \in V(R_1)$. Let R''_1 be the graph formed from R'_1 by placing a diamond on the u_1v_1 edge, and let $f_{R''_1}$ be defined by setting $f_{R''_1}(x) = f_{R'_1}(x) \forall x \in V(R'_1)$ and $f_{R''_1}(x) = 4 - \deg_{R''_1}(x) \forall x \notin V(R'_1)$.

Let R'_2 be the graph formed from R_2^* by relabelling u and v as u_2 and v_2 , respectively, and inserting a new edge between u_2 and v_2 (so u_2v_2 will now be a multi-edge if $uv \in E(R)$). Let $f_{R'_2} = f_{R_2^*}$. Let R''_2 be the graph formed from R'_2 by placing a diamond on the new u_2v_2 edge, and let $f_{R''_2}$ be defined by setting $f_{R''_2}(x) = f_{R'_2}(x) \forall x \in V(R'_2)$ and $f_{R''_2}(x) = 4 - \deg_{R''_2}(x) \forall x \notin V(R'_2)$. Let R'''_2 be the graph formed from R'_2 by instead subdividing the new u_2v_2 edge with a vertex w , and let f'''_2 be defined by $f'''_2(w) = 2$ and $f'''_2(x) = f'_2(x) \forall x \in V(R'_2)$.

Claim 9 (R, f_R) can be satisfied if and only if one of (1), (2) or (3) holds.

Proof The ‘if’ direction follows from a similar ‘gluing’ argument as with case (a)(i), since we can again assume that the appropriate parts of our graphs are drawn in the outside face, so we shall now proceed with proving the ‘only if’ direction:

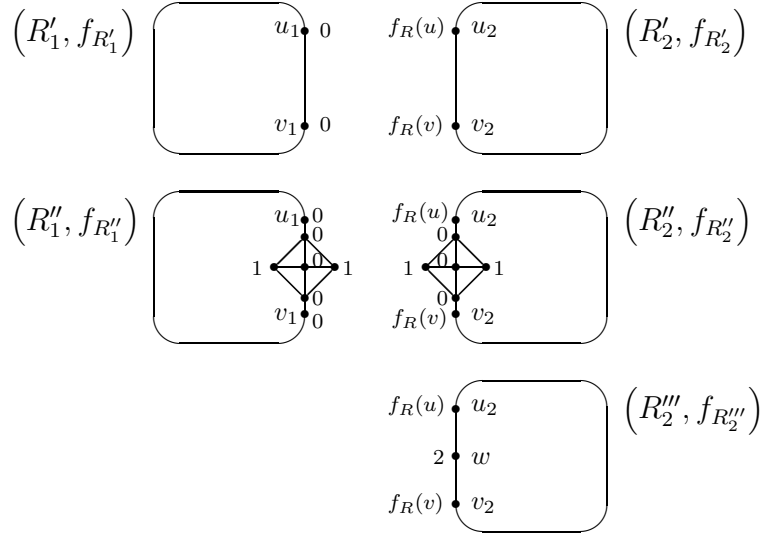


Figure 15: The planar multigraphs R'_1, R''_1, R'_2, R''_2 and R'''_2 , with their discrepancy functions.

Suppose that a plane multigraph M satisfies (R, f_R) . Since $\sum_{x \in V(R_1)} f_R(x)$ and $\sum_{x \in V(R_2^*)} f_R(x)$ are both even, there must be an even number of edges in $E(M) \setminus E(R)$ between $V(R_1)$ and $V(R_2^*)$. As in case (a)(i), these edges must all lie in two faces, so we must either have an even number in both of these faces or an odd number in both. By the same argument as with (a)(i), we may in fact without loss of generality assume that there are either no new edges in both faces or exactly one in both. In the former, it is clear that we can satisfy both $(R'_1, f_{R'_1})$ and $(R'_2, f_{R'_2})$, and in the latter it is clear that we can satisfy both $(R''_1, f_{R''_1})$ and $(R''_2, f_{R''_2})$. Note that we can satisfy $(R'''_2, f_{R'''_2})$ if we can satisfy $(R'_2, f_{R'_2})$ or $(R''_2, f_{R''_2})$. Thus, we can either satisfy $(R'_1, f_{R'_1}), (R'_2, f_{R'_2})$ and $(R'''_2, f_{R'''_2})$, or $(R''_1, f_{R''_1}), (R''_2, f_{R''_2})$ and $(R'''_2, f_{R'''_2})$. In the first case, either (1) or (3) must hold, and in the second case either (2) or (3) must hold. \square

We have now shown that (R, f_R) can be satisfied if and only if (1), (2) or (3) hold. As in case (a)(i), we can use Lemma 7 to determine in $O(|B'_1|^{2.5})$ time whether $(R'_1, f_{R'_1})$ and $(R''_1, f_{R''_1})$ can be satisfied. If neither can be satisfied, we terminate the algorithm. If at least one can be satisfied, then we return to the start of the iterative loop with B'_2 as our new blue graph and either $(R'''_2, f_{R'''_2})$, $(R'_2, f_{R'_2})$ or $(R''_2, f_{R''_2})$ as our augmentation, according to whether both $(R'_1, f_{R'_1})$ and $(R''_1, f_{R''_1})$, just $(R'_1, f_{R'_1})$, or just $(R''_1, f_{R''_1})$ can be satisfied, respectively.

Case (b): $(\overline{u1} \vee \overline{v1}) \wedge u2 \wedge v2$

We shall again start with some groundwork on the structure of R , analogously to case (a), before splitting into subcases.

Since $\overline{u1} \vee \overline{v1}$ holds, we can't have $f(u) = f(v) = 0$. Thus, since $u2 \wedge v2$ also holds, it must be that either u or v has only one edge to B_2 . Hence, since B contains no cut-vertices, it must be that B_2 is connected. Therefore, R_2 must also be connected and so must lie in a single face of R_1^* . Hence, we may proceed in a similar way to case (a), but this time we will split into subcases depending on the parity of R_1^* and R_2 , rather than R_1 and R_2^* .

Case (b)(i): $\sum_{x \in V(R_1^*)} f_R(x)$ and $\sum_{x \in V(R_2)} f_R(x)$ both odd

This time, we let $B'_1 = B_1^* + uv$ (so uv will be a multi-edge in B'_1 if $uv \in E(B)$) and let $B'_2 = B_2^\dagger + uv$. We will define augmentations $(R'_1, f_{R'_1})$ of B'_1 and $(R'_2, f_{R'_2})$ of B'_2 (see Figure 16) such that (R, f_R) can be satisfied if and only if $(R'_1, f_{R'_1})$ and $(R'_2, f_{R'_2})$ can both be satisfied.

Let R'_1 be the graph formed from R_1^* by relabelling u and v as u_1 and v_1 , respectively, and introducing a new vertex w_1 with edges to both u_1 and v_1 . Similarly, let R'_2 be the graph formed from R_2^\dagger by relabelling u and v as u_2 and v_2 , respectively, and introducing a new vertex w_2 with edges to both u_2 and v_2 . Let $f_{R'_1}$ be the discrepancy function on R'_1 defined by setting $f_{R'_1}(u_1) = f_R(u)$, $f_{R'_1}(v_1) = f_R(v)$, $f_{R'_1}(w_1) = 1$, and $f_{R'_1}(x) = f_R(x) \forall x \in V(R_1)$. Let $f_{R'_2}$ be the discrepancy function on R'_2 defined by setting $f_{R'_2}(u_2) = f_{R'_2}(v_2) = 0$, $f_{R'_2}(w_2) = 1$, and $f_{R'_2}(x) = f_R(x) \forall x \in V(R_2)$.

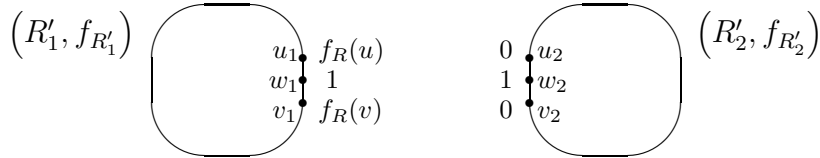


Figure 16: The planar multigraphs R'_1 and R'_2 , with their discrepancy functions.

The proof that (R, f_R) may be satisfied if and only if both $(R'_1, f_{R'_1})$ and $(R'_2, f_{R'_2})$ may be satisfied is as with case (a)(i). Again, we can determine in $O(|B'_1|^{2.5})$ time whether or not $(R'_1, f_{R'_1})$ can be satisfied, and if so we return to the start of the iterative loop with B'_2 as our new blue graph, R'_2 as our new red graph and $f_{R'_2}$ as our new discrepancy function. Otherwise, we terminate the algorithm.

Case (b)(ii): $\sum_{x \in V(R_1^*)} f_R(x)$ and $\sum_{x \in V(R_2)} f_R(x)$ both even

Again, we let $B'_1 = B_1^* + uv$ and $B'_2 = B_2^\dagger + uv$. This time, as with case (a)(ii), we shall define augmentations $(R'_1, f_{R'_1})$ and $(R''_1, f_{R''_1})$ of B'_1 and augmentations

$(R'_2, f_{R'_2}), (R''_2, f_{R''_2})$ and $(R'''_2, f_{R'''_2})$ of B'_2 (see Figure 17) such that (R, f_R) can be satisfied if and only if:

- (1) $(R'_1, f_{R'_1})$ and $(R'_2, f_{R'_2})$ can both be satisfied, but $(R'_1, f_{R''_1})$ can't;
- (2) $(R'_1, f_{R''_1})$ and $(R''_2, f_{R''_2})$ can both be satisfied, but $(R'_1, f_{R'_1})$ can't; or
- (3) $(R'_1, f_{R'_1}), (R'_1, f_{R''_1})$ and $(R''_2, f_{R''_2})$ can all be satisfied.

Let R'_1 be the graph formed from R_1^* by relabelling u and v as u_1 and v_1 , respectively, and inserting an edge between u_1 and v_1 (so u_1v_1 will now be a multi-edge if $uv \in E(R)$). Let $f_{R'_1}$ be defined by setting $f_{R'_1}(x) = f_R(x) \forall x \in V(R'_1)$. Let R''_1 be the graph formed from R'_1 by placing a diamond on the u_1v_1 edge, and let $f_{R''_1}$ be defined by setting $f_{R''_1}(x) = f_{R'_1}(x) \forall x \in V(R'_1)$ and $f_{R''_1}(x) = 4 - \deg_{R''_1}(x) \forall x \notin V(R'_1)$.

Let R'_2 be the graph formed from R_2^\dagger by relabelling u and v as u_2 and v_2 , respectively, and inserting a new edge between u_2 and v_2 . Let $f_{R'_2}$ be the discrepancy function on R'_2 defined by setting $f_{R'_2}(u_2) = f_{R'_2}(v_2) = 0$ and $f_{R'_2}(x) = f_R(x) \forall x \in V(R_2)$. Let R''_2 be the graph formed from R'_2 by placing a diamond on the new u_2v_2 edge, and let $f_{R''_2}$ be defined by setting $f_{R''_2}(x) = f_{R'_2}(x) \forall x \in V(R'_2)$ and $f_{R''_2}(x) = 4 - \deg_{R''_2}(x) \forall x \notin V(R'_2)$. Let R'''_2 be the graph formed from R'_2 by instead subdividing the new u_2v_2 edge with a vertex w , and let $f_{R'''_2}$ be defined by $f_{R'''_2}(w) = 2$ and $f_{R'''_2}(x) = f_{R'_2}(x) \forall x \in V(R'_2)$.

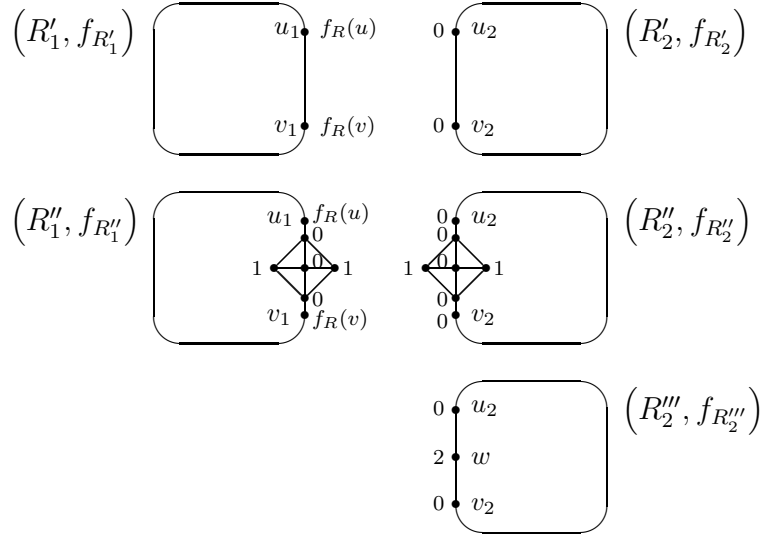


Figure 17: The planar multigraphs R'_1, R''_1, R'_2, R''_2 and R'''_2 , with their discrepancy functions.

The proof that (R, f_R) can be satisfied if and only if (1),(2) or (3) hold is as with case (a)(ii). Again, we can determine in $O(|B_1|^{2.5})$ time whether or not $(R'_1, f_{R'_1})$ and $(R''_1, f_{R''_1})$ can be satisfied, and if at least one can then we return

to the start of the iterative loop with B'_2 as our new blue graph and either $(R'''_2, f_{R'''_2})$, $(R'_2, f_{R'_2})$ or $(R''_2, f_{R''_2})$ as our augmentation, according to whether both $(R'_1, f_{R'_1})$ and $(R''_1, f_{R''_1})$, just $(R'_1, f_{R'_1})$, or just $(R''_1, f_{R''_1})$ can be satisfied, respectively. If neither $(R'_1, f_{R'_1})$ nor $(R''_1, f_{R''_1})$ can be satisfied, we terminate the algorithm.

Case (c): $(\overline{u1} \vee \overline{v1}) \wedge (\overline{u2} \vee \overline{v2})$

We will now deal with the remaining case, which will follow from a detailed investigation of the properties that are forced upon us if $(\overline{u1} \vee \overline{v1}) \wedge (\overline{u2} \vee \overline{v2})$ holds.

Recall that if we have $\overline{u1}$, then by definition $f_R(u) \geq 1$ and u has at least two edges to R_1 , so u must have only one edge to R_2 , and hence we have $u2$. Similarly, $\overline{v1} \Rightarrow v2$, $\overline{u2} \Rightarrow u1$ and $\overline{v2} \Rightarrow v1$. Thus, the only possibilities are $u1 \wedge \overline{u2} \wedge \overline{v1} \wedge v2$ and $\overline{u1} \wedge u2 \wedge v1 \wedge \overline{v2}$. By swapping u and v if necessary, we can without loss of generality assume that we have the former.

Note that the only way to obtain $u1 \wedge \overline{u2}$ is to have exactly one edge in B from u to B_1 (or, equivalently, exactly one edge in R from u to R_1), exactly two edges in B from u to B_2 , no edges in B from u to v , and $f_R(u) = 1$. Similarly, we must have exactly one edge in B from v to B_2 , exactly two edges in B from v to B_1 , and $f_R(v) = 1$. Note also that we must have $|B_1| = 1$, since otherwise the minimality of B_1 would imply that u and v would both have to have at least two edges in B to B_1 , which would in turn imply that we would have to have $u2 \wedge v2$. Thus, B must be as shown in Figure 18.

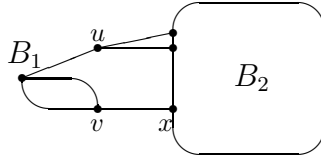


Figure 18: The structure of B in case (c).

If $|B_2| = 1$, then $|R|$ is bounded by a constant and so we can determine the satisfiability of (R, f_R) in $O(1)$ time (simply by checking all graphs with $|R|$ vertices to see if any of these do satisfy (R, f_R)).

If $|B_2| > 1$, then let x denote the neighbour of v in B_2 , let $\widehat{B}_1 = B_1 \cup v$ and let $\widehat{B}_2 = B_2 \setminus x$. Note that ux forms a 2-vertex-cut where u and x both have just one edge to \widehat{B}_1 (see Figure 19). Hence, we can copy case (a) with B_1 and B_2 replaced by \widehat{B}_1 and \widehat{B}_2 , respectively, to again obtain graphs B'_1 and B'_2 and appropriate augmentations. It may be that the graph B'_1 will have a 2-vertex-cut, so this time we won't be able to use Lemma 7 to determine the satisfiability of augmentations of it. However, we know that we will have $|B'_1| = 4$, so the number of vertices in any augmentation of B'_1 will be bounded by a constant, and hence we will be able to determine satisfiability of these augmentations in $O(1)$ time.

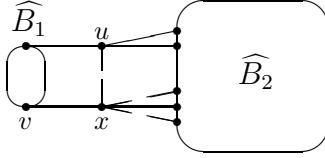


Figure 19: The 2-vertex-cut $\{u, x\}$.

Running Time

We shall now show that the algorithm takes $O(|H|^{2.5})$ time. It is fairly easy to see that the first three stages can be accomplished within this limit (in fact, they take only $O(|H|^2)$ time), so we will proceed straight to an examination of Stage 4.

We apply Stage 4 to each of the 2-vertex-connected blocks derived from Stage 3. It is easy to see that the total number of vertices in all these blocks is at most $2|H|$, since each vertex of H will only appear in at most two of these, so it will actually suffice just to deal with the case when H is itself a 2-vertex-connected block, i.e. when we start Stage 4 with only one 2-vertex-connected block, and it has $|H|$ vertices.

During each iteration of Stage 4, we take a graph B and use it to construct graphs B'_1 and B'_2 , where $|B'_1| + |B'_2| = |B| + 2$ and $|B'_2| < |B|$, before replacing B with B'_2 and iterating again. Let $B'_{1,1}, B'_{1,2}, \dots, B'_{1,l}$, for some l , denote the various graphs that take the role of B'_1 during our algorithm. Since $|B'_2| < |B|$, we can only have at most $|H|$ iterations, and so we must have $\sum_i |B'_{1,i}| \leq 3|H|$ (by telescoping, since we always have $|B'_1| + |B'_2| = |B| + 2$). We need to apply the algorithm given by Lemma 7 to at most three augmentations of each $B'_{1,i}$, so the total time taken by all such applications will be at most $3\lambda \sum_i (|B'_{1,i}|^{2.5}) \leq 3\lambda (\sum_i |B'_{1,i}|)^{2.5} = O(|H|^{2.5})$.

At the start of each iteration, we wish to determine whether B has any 2-vertex-cuts and, if so, find a minimal one. Using an algorithm from [8] for decomposing a graph into its so-called ‘triconnected components’, this takes $O(|B|) = O(|H|)$ time. It is fairly clear that all other operations involved in an iteration of Stage 4, aside from applications of Lemma 7, can also be accomplished within $O(|H|)$ time, so (since we recall that there are at most $|H|$ iterations) this all takes $O(|H|^2)$ time in total (in fact, by careful bookkeeping, this could be reduced to $O(|H|)$). Hence, it follows that the whole algorithm takes $O(|H|^{2.5})$ time.

Comments

By keeping track of all the operations, the algorithm can be used to find an explicit 4-regular planar multigraph $G \supset H$ if such a graph exists, also in $O(|H|^{2.5})$ time. If H is simple, then we can also obtain a 4-regular simple planar graph $G' \supset H$ without affecting the order of the overall running time,

using the proof of Lemma 2.

Aside from looking for a perfect matching during the applications of Lemma 7, every part of the algorithm can be accomplished in $O(|H|^2)$ time. It would therefore be interesting to know if the special structure of the graphs seen in Lemma 7 could be exploited to obtain a faster perfect matching algorithm.

References

- [1] M. Bodirsky, C. Gröpl, *Generating labelled planar graphs uniformly at random*, Theoretical Computer Science **379** (2007), 377-386.
- [2] H.L. Bodlaender, R.B. Tan, J. van Leeuwen, *Finding a Δ -regular supergraph of minimum order*, Discrete Applied Mathematics **131** (2007), 3-9.
- [3] K.S. Booth, G.S. Lueker, *Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms*, Journal of Computer and System Sciences **13** (1976), 335-379.
- [4] C. Dowden, *Uniform random planar graphs with degree constraints*, DPhil thesis (2008), available at <http://ora.ouls.ox.ac.uk>.
- [5] S. Even, O. Kariv, *An $O(n^{2.5})$ algorithm for maximum matching in graphs*, Proceedings of the 16th Annual IEEE Symposium on Foundations of Computer Science (1975), 100-112.
- [6] S. Gerke, C. McDiarmid, A. Steger, A. Weißl, *Random planar graphs with n nodes and a fixed number of edges*, Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (2005), 999-1007.
- [7] O. Giménez, M. Noy, *Asymptotic enumeration and limit laws of planar graphs*, Journal of the American Mathematical Society **22** (2009), 309 - 329.
- [8] J.E. Hopcroft, R.E. Tarjan, *Dividing a graph into triconnected components*, SIAM Journal on Computing **2** (1973), 135-158.
- [9] C. McDiarmid, A. Steger, D. Welsh, *Random planar graphs*, Journal of Combinatorial Theory Series B **93** (2005), 187-205.
- [10] H. Whitney, *2-isomorphic graphs*, American Journal of Mathematics **55** (1933), 245-254.